Cost-optimization of many-robot systems

Douglas W. Gage

RDT&E Division, Naval Command Control and Ocean Surveillance Center
NCCOSC RDTE DIV 531, San Diego, CA 92152-7383

ABSTRACT

Cost optimization will be essential to the economic viability of future many-robot systems. Unfortunately, cost-optimal design will often depend sensitively on the performance and costs of candidate sensor and other components, and on both explicit mission-specific and implicit common sense application requirements. The development of a standardized integration architecture providing multi-level protocols to support the flexible but tight integration of off the shelf subsystem components into low cost robots will facilitate the development of this soon to emerge class of systems.

1. INTRODUCTION

The rapid evolution of micromechanical fabrication techniques and other enabling technologies suggests that systems consisting of large numbers of simple autonomous robots may soon provide an appropriate solution to many real world applications [1]. The colonies of the social insects (ants, termites and bees) serve as outstanding models of functioning systems consisting of large numbers of quasi-intelligent mobile elements. Through experimental manipulation of insect colonies and computer simulations, researchers have elucidated some of the mechanisms which these colonies employ to survive and grow by adapting to their changing environment [2,3,4]. With various individual and group animal behaviors serving as "existence proofs", quasi-intelligent "emergent behavior" resulting from the interaction of simple reactive planners has been proposed as the basis for the intelligent control of individual robots [5]. The term "Swarm Intelligence" has been used to describe the application of this approach to distributed systems consisting of perhaps hundreds of elements [6,7]. Biological models are explicitly acknowledged as the motivation for much of this work [8,9].

In previous papers we have developed (a) the notion of “coverage” as a paradigm for the system level functionality of many robot systems [10], (b) some initial notions of sensor-based behaviors to implement various coverage modes [10,12], (c) measures of effectiveness and system design considerations for the generic area search application (e.g., minesweeping) [11,12], and (d) an approach to the communications needs of a system consisting of an arbitrarily large number of simple autonomous robots [13].

In this paper we examine some of the factors that will determine how we can most successfully develop actual systems comprising hundreds, thousands, or millions of robots. In Section 2 we consider some of the factors that guide the "evolution" of artificial systems in general, and focus on the necessity of cost efficiency in implementing the elements of many-robot systems in particular. Section 3 demonstrates that the optimal way to implement element behaviors will depend <u>sensitively</u> on the system resources available and on the detailed requirements of specific applications. Section 4 argues that the development of a standardized <u>integration</u> architecture based on LAN technology and multilayered protocols will prove critical to the cost effective development of many of these systems.

2. COST EFFICIENCY AS A DRIVER FOR THE "EVOLUTION" OF ARTIFICIAL SYSTEMS

Just as nature provides examples of many-element natural systems (notably the social insects), so she provides an example of process in natural evolution. Why is it that some species have come into existence, while others, apparently equally plausible, have not? In order to actually exist, a species of natural organism must be "possible" in at least the following four senses: (a) <u>physiological</u>: the organism's body must be capable of functioning in a coherent fashion in order to sustain its living existence to the point that it can have offspring; (b) <u>ontogenetic</u>: it must be capable of developing from an embryo; (c) <u>phylogenetic</u>: its genotype must be derivable from its progenitors by the natural "genetic algorithms" of sex and mutation; and (d) <u>ecological</u>: it must be able to survive and reproduce in the environment in which it is situated.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
NOV 1994	**N/A**	**-**

4. TITLE AND SUBTITLE	
Cost-Optimization of Many-Robot Systems	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Navy Command, Control and Ocean Surveillance Center Division 531 RDT&E San Diego, CA 92152-7383**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**UU**	**7**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

(Humans have of course affected these rules of the game through the domestication of plants and animals, selective breeding, grafting of plants, and genetic engineering.)

These existence criteria for natural systems are reflected in analogous existence criteria for artificial systems: (a) functional: the system must be capable of performing the function for which it has been designed; (b) manufacturable: the system must be capable of being manufactured by existing manufacturing processes; (c) designable: the design of the system must be imaginable by designers working in their cultural context; and (d) marketable: the system must be perceived to serve some purpose well enough, when compared to competing approaches, to warrant its design and manufacture.
Appropriate technological, economical, and cultural contexts are thus prerequisite to the development of any new product. For systems consisting of large numbers of small robots, the rapid advancement of MEMS and other robotics-relevant technologies opens up great opportunities by expanding the envelopes of functionality and manufacturability. The key challenge is to conceive and design systems that can be successfully implemented and sold as viable products in a competitive marketplace.

For many-robot systems, even more than most other products, cost efficiency is both necessary and, in terms of the economics of the development process, feasible: the fact that a system consists of a large number of identical units places a premium on lowering the cost of each unit, and, in fact, justifies significant "up front" investment for this purpose, since this investment is amortized over many units. If the initial entry into a viable market segment is not cost-optimized, it will soon be displaced by later entries which are. The problem is, of course, how to realize a cost-optimized design.

3. FACTORS DETERMINING COST-OPTIMAL SYSTEM DESIGN

The designer of a many-robot system has a rich design space to explore in the quest for a cost optimal system. In some applications, a large number of the simplest possible elements may be the right answer; in other applications the best solution may be a much smaller number of elements incorporating higher capability sensors, effectors, processing and/or communications resources. The elements may exhibit simple independent behaviors, or complex coordinated ones. And superficially similar behaviors (both individual and ensemble) can often be implemented in a number of very different ways. In this section we consider a number of the factors that determine which of several possible design approaches will be most cost effective.

3.1 Sensitivity to Subsystem Price and Performance

Relatively minor changes in the price and performance level of various sensor and effector subsystems can lead to qualitative differences in overall system design. As an example, let us consider an area search application such as minesweeping or lost object retrieval which can be addressed with either a coordinated search strategy (such as a classical "lawnmower" search pattern), or with a presumably less expensive randomized search strategy. The coordinated search is optimal in the sense that the searchers are always searching in the places where the target(s) are most likely to be found (i.e., the places which have been previously searched the least), while the random searchers search some spots many times before covering some others even once. We can quantify this difference in effectiveness by calculating the fraction of targets detected (D) as a function of the sensor's probability of detecting a target within its range (p) and the equivalent number of times we have "swept" the area (S). We find, for the completely coordinated search model:

$$D_c = 1 - (1 - p)^{Sc} \tag{1}$$

while the random search model yields:

$$D_r = 1 - e^{- p\ Sr} \tag{2}$$

The coordinated search strategy's advantage in effectiveness per unit of search effort can be traded off against the random strategy's advantage in cost per unit of search effort to determine the lowest system cost. As figure 1 indicates, the bottom line choice between the two strategies depends sensitively on the relative increase in cost required to achieve the coordinated search (as a fraction of the baseline element cost) and the quality of the target detection sensor (probability of detection of a

target within the nominal sensor range). This latter dependency is due to the fact that the effectiveness penalty for searching an area you have searched already is lower if the probability of having missed any targets there is higher. See references [11,12] for additional details of the analysis and for presentation of a "toy" design exercise in which changing sensor price and performance levels caused the number of robots in a cost-optimized multi-target search system to vary from 50 to 461 -- a factor of more than 9 to 1.

This sensitivity of overall system design to subsystem price performance is especially critical because micro-electro-mechanical systems (MEMS) and other sensor and effector technologies are evolving so rapidly; it is quite likely that advances in technology could occur during the development cycle of a many-robot system which would justify a complete change of approach to the system design.

3.2 Sensitivity to Details of Explicit Mission-Specific Goals

It is sometimes the case that the optimal design of a system will depend sensitively on specific details of the mission goals. For example, the tradeoff between coordinated and random area search strategies is different when (as in clearing a minefield) the mission goal is to find a specified percentage of targets with minimum cost (upper curve on figure 1) than when (as in lost object recovery) it is to find a single target in an area with minimum cost (lower curve). Perhaps surprisingly, if the presence of a single target is suspected but not certain, and it is desired to achieve a specified confidence that it is in fact not present (given that it has not been found), then the <u>upper</u> curve applies.

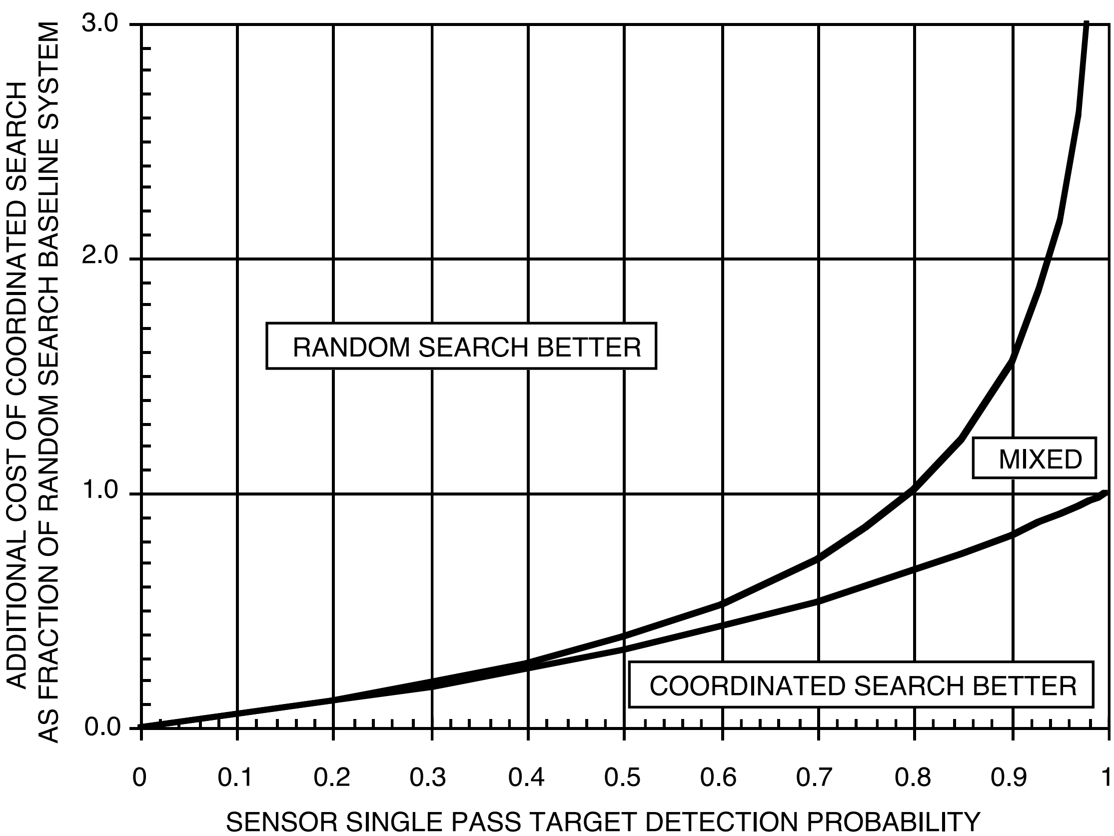

Figure 1. System cost-based tradeoff between coordinated and random search models as function of (a) additional cost of coordinated search as fraction of random search baseline system, (b) target detection probability of search sensor, and (c) explicit mission goal. Above the upper line, random search is lower cost for both single target and multi-target search. Below the lower line, coordinated search is better for both. In the intermediate region (marked "mixed") random search is better for finding a single target, coordinated search is better for multiple targets. See references [11,12] for details of analysis.

3.3 Sensitivity to Details of Implicit Policies

The best choice of implementation of various ensemble behaviors may depend sensitively on high level criteria which may have never been explicitly acknowledged. For example, consider an air-deployed military system in which a herd of 10,000 elements are to configure themselves in a uniform density filled-circular pattern 1 km in diameter, centered on a target beacon. Suppose that one of the two 5,000 element canisters fails in deployment -- what do we want the remaining 5,000 elements to do? One obvious possibility would be to maintain the spacing of the elements, but reduce the diameter of coverage to about 700 meters. Another obvious possibilitiy is to maintain the 1 km diameter of the coverage, but increase the average linear spacing between elements by about 40%. Or we could split the difference. The point is that, while this call is a "high level" policy decision that might conceivably change from mission to mission, it has implications for the design of a "low level" element condensation algorithm. If the element density is to stay constant, then the algorithm might be something like "if local density of elements is too low, move toward beacon; otherwise follow the density gradient", while if it is the diameter which is to stay constant, it might be something like "if my distance to the beacon is greater than half the specified diameter, move toward beacon; otherwise follow the density gradient".

Even the least dependable human agent embodies a vast amount of implicit understanding about the world which will be lacking in our robotic systems, as well as an ability to understand (if not obey) changes made to previously established plans. If our robotic system ensemble behavior is to be flexibly adaptive to its environment, and not unacceptably "brittle", it will be necessary to *explicitly* formulate and implement policies to deal with many contingencies that a human would deal with by using "common sense", and these policies will often be hardwired into the system -- that is, the mechanisms implementing the policies may not be capable of being reprogrammed in the field in order to support alternative policies, should they be desired.

3.4 Sensitivity to Available Mission-Specific Resources

A large robot is sometimes conveniently implemented as a generic mobility base (typified by the Cybermotion and TRC indoor products, and the ARPA/Navy UUV in the underwater domain) that provides "bus" (mobility and navigation) and "hotel" (space, environmental conditioning, power, communication) support to one or more distinctly separate mission-packages. For the small robots comprising the systems we are considering, however, efficiency will dictate that sensor and/or effector subsystem resources incorporated to support the functional and/or performance requirements of a specific mission should also be fully exploited to support generic navigation and/or communication behaviors which otherwise would be implemented in other ways, or not implemented at all -- the line between "platform generic" and "mission package specific" functions will become increasingly blurred.

4. RECONCILING DESIGN FLEXIBILITY AND EFFICIENCY

The last section demonstrated that the lowest cost approach to the implemention of desired behaviors in many-robot systems will depend sensitively on both the relative costs and capabilities of candidate sensors and effectors and the detailed requirements of the intended application. The system designer therefore needs a highly flexible integration paradigm that can quickly and easily support the "evolution" of robots to exploit the differing combinations of subsystems needed to address the requirements of an expanding diversity of applications, as well as to accommodate continuing rapid advances in component technologies.

As Flynn argued in [1], the real payoff from tiny robots will come from "tight" physical integration of components: size, weight and power requirements will have to be minimized in order to minimize cost. Cables, connectors, and duplicated resources will be primary targets for elimination. The ultimate goal is monolithic fabrication of complete robots in a fashion analogous to (and perhaps nearly identical to) the manufacture of integrated circuits. Electronic Design Automation (EDA) tools for the development of ASICs offer a useful model of what can be done to support the development of complex, highly integrated systems, and the process of extending these tools to MEMS processes and designs has begun.

Even realizing the dream of monolithic MEMS fabrication of micro-robots, however, will not solve all our integration problems. Many applications will continue to require macro-sized robots, and the growing diversity of relevant sensor and effector technologies implies an ever growing requirement to integrate more or less "off the shelf" sensors into robots, for

singly-deployed robots as well as for many-robot systems. The problem is to reconcile the need for lightweight, compact, inexpensive, high bandwidth, "tight" integration of subsystems with the need for flexibility in configuring systems from diverse subsystem elements.

4.1 Efficient Integration Architecture

The word "architecture" is often used in the robotics community in the sense of "intelligent control architecture"; i.e., as a mechanism for implementing desired behaviors of effectors based on inputs from sensors. Examples are Brooks's subsumption architecture [5] and Albus's RCS [14]. An integration architecture, on the other hand, is a mechanism for gluing predefined subsystems together into a coherently functioning whole. (Yet another use of the word is in the sense of implementation architecture, as a style of designing a system and building it -- one central processor vice a number of smaller ones.)

Unfortunately, sensor subsystems on the market exhibit a dizzying variety of interfaces for communicating data, control, and status: analog, discrete binary digital, PCM digital, parallel digital, serial digital, servo control, relay closure, and mixtures of the above. Sometimes a vendor decides to provide a "complete subsystem solution" by adding a microcontroller and providing the user with an asynchronous RS-232 link. Products available with an RS-232 interface include video cameras, pan/tilt mechanisms, laser rangefinders, and radar track processors. RS-232 has several serious deficiencies, however: (a) it is a point-to-point link which connects one device to one other device, so that using many devices requires many serial ports; (b) it is slow, imposing long message latency time and providing limited bandwidth; (c) it is expensive to use since it is necessary to parse the incoming character stream to extract meaningful data; and (d) it is asynchronous, so that the arrival of a complete message can only be determined by a succssful message parse, and it is difficult to reliably detect a link failure.

Fortunately, LAN technology is now readily available to solve many of the basic communications deficiencies of RS-232, connecting multiple nodes with high speed synchronous communications. The OSI Reference Model [15] supplements basic LAN technology with a protocol framework which could be employed to represent robot-specific software constructs. Off-the-shelf sensor or effector subsystems with LAN interfaces are still, however, rare. Some integrators of large robotic (and telerobotic) systems have provided a LAN interface for each subsystem by incorporating an additional processing element (variously termed a Front End Processor, Network Front End, Interface Processor, or Intelligent Communications Interface [16]). Clearly, this approach is at variance with "optimal" per-system cost-minimization; however, continuing advances in VLSI technology allows the implementation of microcontroller elements costing only a few dollars in volume production. A number of candidate network architectures for distributed control have emerged in recent years to address such application areas as building control (including the "Smart House" concept) and integrated automotive electronics [17], and the IEEE and NIST have undertaken a joint effort to develop a communications standard for "smart sensors" [18]. Echelon's LonWorks architecture in particular seems to offer a number of valuable features to support robotic integration: typed network variables, a robust protocol stack, a choice of diverse LAN media, and protocol extensibility.

5. CONCLUSIONS

Marketplace realities will dictate that many-robot systems (and also other mass-produced robots sold one at a time) must be highly cost-optimized, implying a tight physical integration of sensor, effector, and processing component subsystems. The up-front investment necessary to make these systems cost efficient may tend to inhibit the entry of many-robot systems into new applications. When possible, it will pay to "piggy back" on systems previously developed for other applications, or on mass market items which have been developed (and cost-optimized) for other purposes, such as the toy market.

While small companies with vision and imagination may pioneer the development of many-robot systems, larger companies with the resources to cost-optimize and mass-produce the products may ultimately dominate the marketplace. Because cost-optimal design will often depend sensitively on the performance and costs of candidate sensor and other components, and on both explicit mission-specific and implicit common sense application requirements, a wide variety of control schemes and behaviors will have to be developed. Unfortunately, the need for cost efficient tight physical integration will tend to make rapid prototyping difficult, complicating this evolution.

The development of a "standard" integration architecture, capable of supporting many different paradigms for intelligent control, would ameliorate this problem, just as the introduction of the IBM PC and MS-DOS provided a standard framework within which independent developers of both hardware subsystems and software could implement their ideas. Such an integration architecture might incorporate aspects of LAN-based protocol stacks and EDA tools for ASIC development. The payoff will be to facilitate the exciting exploration of a very highly dimensioned design space, leading to the discovery of clever tricks for the efficient and effective use of hardware resources. And this is the ultimate lesson offered by natural evolution: the shameless exploitation of discovered opportunity.

6. ACKNOWLEDGEMENTS

This work is supported by the Intelligent Systems Program of the Office of Naval Research, Computer Science Division.

7. REFERENCES

1. Flynn, A.M. "Gnat Robots (And How They Will Change Robotics)", **Proceedings of the IEEE Micro Robots and Teleoperators Workshop**, Hyannis MA, 9-11 November 1987. Also appeared in **AI Expert**, December 1987, p 34 et seq.

2. Deneubourg, J.L., et al. "The Dynamics of Collective Sorting Robot-Like Ants and Ant-Like Robots", **Proceedings of First International Conference on Simulation of Adaptive Behaviors**, pp 356-363.

3. Franks, N.R. "Army Ants: A Collective Intelligence", **American Scientist**, vol 77, March-April 1989, pp 139-145.

4. Seeley, T.D. "The Honey Bee Colony as a Superorganism", **American Scientist**, vol 77, November-December 1989, pp 546-553.

5. Brooks, R.A. "A Robust Layered Control System for a Mobile Robot", **IEEE Journal of Robotics and Automation**, RA-2, April 1986, pp 14-23.

6. Beni, G., and Wang, J. "Theoretical Problems for the Realization of Distributed Robotic Systems", **Proceedings of the 1991 IEEE International Conference on Robotics and Automation**, Sacramento CA, April 1991, pp 1914-1920.

7. Hackwood, S. and Beni, G. "Self-organization of Sensors for Swarm Intelligence", **Proceedings of the 1992 IEEE International Conference on Robotics and Automation**, Nice France, May 1992, pp 819-829.

8. Arkin, R.C. and Hobbs, J.D. "Dimensions of Communication and Social Organization in Multi-agent Robotic Systems", **Proceedings of the 2d Conference on the Simulation of Adaptive Behavior: From Animals to Animats (SAB92)**, Honolulu HI, December 1992, pp 486-493.

9. Kube, C.R., and Zhang, H. "Collective Robotic Intelligence", **Proceedings of the 2d Conference on the Simulation of Adaptive Behavior: From Animals to Animats (SAB92)**, Honolulu HI, December 1992, pp 460-468.

10. Gage, D.W. "Command Control for Many-Robot Systems", **AUVS-92, the Nineteenth Annual AUVS Technical Symposium**, Huntsville AL, 22-24 June 1992. Reprinted in **Unmanned Systems** Magazine, Fall 1992, Volume 10, Number 4, pp 28-34.

11. Gage, D.W. "Sensor Abstractions to Support Many-Robot Systems", **Proceedings of SPIE Mobile Robots VII**, Boston MA, 18-20 November 1992, Volume 1831, pp 235-246.

12. Gage, D.W. "Randomized Search Strategies with Imperfect Sensors", **Proceedings of SPIE Mobile Robots VIII**, Boston MA, 9-10 September 1993, Volume 2058.

13. Gage, D.W. "How to Communicate with Zillions of Robots", **Proceedings of SPIE Mobile Robots VIII**, Boston MA, 9-10 September 1993, Volume 2058.

14. Albus, J.S. "System Description and Design Architecture for Multiple Autonomous Undersea Vehicles", NIST Technical Note 1251, September 1988.

15. International Organization for Standards (ISO), "Information Processing Systems - Open Systems Interconnection - Basic Reference Model", ISO 7498:1984.

16. Gage, D.W., Harmon, S.Y., Aviles, W.A., and Bianchini, G.L. "Subsystem Interfaces for Complex Robots", **Proceedings of Workshop on Robot Standards**, Detroit MI, 6-7 June 1985, pp 119-124.

17. Raji, R.S. "Smart networks for control", **IEEE Spectrum**, June 1994, pp 49-55.

16. Bryzek, J. "A Communications Standard for Low-Cost Sensors", **Sensors**, September 1994, pp 18-24.